

Oracle® Fusion Middleware

Understanding Oracle GoldenGate



19c (19.1.0)

E98077-01

April 2019

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Oracle Fusion Middleware Understanding Oracle GoldenGate, 19c (19.1.0)

E98077-01

Copyright © 2018, 2019, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

1 Introduction to Oracle GoldenGate

What is Oracle GoldenGate?	1-1
Why Do You Need Oracle GoldenGate?	1-1
When Do You Use Oracle GoldenGate?	1-2
Topologies for Oracle GoldenGate	1-3
Oracle GoldenGate Product Family	1-3

2 Getting Started with Oracle GoldenGate

Oracle GoldenGate Supported Processing Methods and Databases	2-3
Components of Oracle GoldenGate Microservices Architecture	2-4
What is a Service Manager?	2-7
What is an Administration Server?	2-7
What is a Receiver Server?	2-8
What is a Distribution Server?	2-10
What is a Performance Metrics Server?	2-10
What is the Admin Client?	2-11
Roadmap for Implementing the Microservices Architecture	2-12
Components of Oracle GoldenGate Classic Architecture	2-13
What is a Manager?	2-14
What is a Data Pump?	2-14
What is a Collector?	2-15
What is GGSCI?	2-16
Common Data Replication Processes	2-16
What is an Extract?	2-17
What is a Trail?	2-17
What is a Replicat?	2-19

3 Oracle GoldenGate Processes and Key Terms

Overview of Target-Initiated Paths	3-1
Oracle GoldenGate Key Terms and Concepts	3-1
Overview of Process Types	3-2

Overview of Groups	3-2
Overview of the Commit Sequence Number (CSN)	3-2

Preface

Understanding Oracle GoldenGate describes data replication concepts, Oracle GoldenGate Classic Architecture and Oracle GoldenGate Microservices Architecture, and their components. Using these concepts, you can plan a data replication solution with Oracle GoldenGate.

- [Audience](#)
- [Documentation Accessibility](#)
- [Conventions](#)
- [Related Information](#)

Audience

This guide is intended for system administrators or application developers who need to learn about Oracle GoldenGate concepts. It is assumed that readers are familiar with web technologies and have a general understanding of Windows and UNIX platforms.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Related Information

The Oracle GoldenGate Product Documentation Libraries are found at

<https://docs.oracle.com/en/middleware/goldengate/index.html>

Additional Oracle GoldenGate information, including best practices, articles, and solutions, is found at:

[Oracle GoldenGate A-Team Chronicles](#)

1

Introduction to Oracle GoldenGate

Learn about Oracle GoldenGate concepts, why and when should you use it, and get familiar with some of the basic terminology and keywords associated with Oracle GoldenGate.

Topics:

- [What is Oracle GoldenGate?](#)
Oracle GoldenGate is a comprehensive software package for real-time data integration and replication. It enables high availability solutions, real-time data integration, transactional change data capture, data replication, transformations, and verification between operational and analytical enterprise systems.
- [Why Do You Need Oracle GoldenGate?](#)
Enterprise data is typically distributed across the enterprise in heterogeneous databases. To get data between different data sources, you can use Oracle GoldenGate to load, distribute, and filter transactions within your enterprise in real-time and enable migrations between different databases in near zero-downtime.
- [When Do You Use Oracle GoldenGate?](#)
Oracle GoldenGate meets almost any data movement requirements you might have. Some of the most common use cases are described in this section.
- [Topologies for Oracle GoldenGate](#)
After installation, Oracle GoldenGate can be configured to meet your organization's business needs.
- [Oracle GoldenGate Product Family](#)
There are numerous products in the Oracle GoldenGate product family.

What is Oracle GoldenGate?

Oracle GoldenGate is a comprehensive software package for real-time data integration and replication. It enables high availability solutions, real-time data integration, transactional change data capture, data replication, transformations, and verification between operational and analytical enterprise systems.

Using Oracle GoldenGate, you can move committed transactions across multiple systems in your enterprise. Oracle GoldenGate enables you to replicate data between Oracle databases to other supported heterogeneous database, and between heterogeneous databases. In addition, you can replicate to Java Messaging Queues, Flat Files, and to Big Data targets in combination with Oracle GoldenGate for Big Data. To know more, see <https://www.oracle.com/middleware/technologies/goldengate.html>.

Why Do You Need Oracle GoldenGate?

Enterprise data is typically distributed across the enterprise in heterogeneous databases. To get data between different data sources, you can use Oracle GoldenGate to load, distribute, and filter transactions within your enterprise in real-time and enable migrations between different databases in near zero-downtime.

To do this, you need a means to effectively move data from one system to another in real-time and with zero-downtime. Oracle GoldenGate is Oracle's solution to replicate and integrate data.

Oracle GoldenGate has the following key features:

- Data movement is in real-time, reducing latency.
- Only committed transactions are moved, enabling consistency and improving performance.
- Different versions and releases of Oracle Database are supported along with a wide range of heterogeneous databases running on a variety of operating systems. You can replicate data from an Oracle Database to a different heterogeneous database.
- Simple architecture and easy configuration.
- High performance with minimal overhead on the underlying databases and infrastructure.

When Do You Use Oracle GoldenGate?

Oracle GoldenGate meets almost any data movement requirements you might have. Some of the most common use cases are described in this section.

You can use Oracle GoldenGate to meet the following business requirements:

Business Continuity and High Availability

Business Continuity is the ability of an enterprise to provide its functions and services without any lapse in its operations. High Availability is the highest possible level of fault tolerance. To achieve business continuity, systems are designed with multiple servers, multiple storage, and multiple data centers to provide high enough availability to support the true continuity of the business. To establish and maintain such an environment, data needs to be moved between these multiple servers and data centers, which is easily done using Oracle GoldenGate.

Consider a scenario where you are working in a multinational bank that has its headquarters in London, UK. You work in one of the banks' branches in Bangalore, India. This bank uses a specific account for its financial application that is used globally at all the branches. You have been asked by your manager to daily synchronize the transactions that have happened for this account in the database in the Bangalore branch with the centralized database situated at the UK. The volume of transactions is massive, and even the slightest delay can greatly impact the business. This same process is required at multiple destinations for every database in all the branches of the bank worldwide. This process has to be monitored continuously, preferably through some sort of GUI-based tool for the ease of management. Additionally, the bank has several other, non-critical applications used at all the branches. These applications are based on heterogeneous databases, such as MySQL, but the transactions done over these databases also must be loaded into an Oracle Database located at the headquarters. The replication technology used must support both Oracle and heterogeneous databases so that they can talk to each other. Oracle GoldenGate is an apt solution in such a scenario.

Initial Load and Database Migration

Initial load is a process of extracting data records from a source database and loading those records onto a target database. Initial load is a data migration process that is performed only once. Oracle GoldenGate allows you to perform initial load data migrations without taking your systems offline.

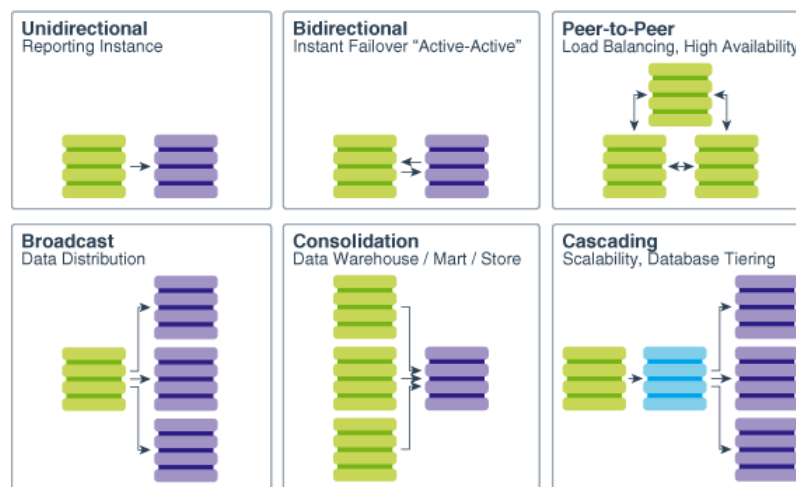
Data Integration

Data integration involves combining data from several disparate sources, which are stored using various technologies, and provide a unified view of the data. Oracle GoldenGate provides real-time data integration.

Topologies for Oracle GoldenGate

After installation, Oracle GoldenGate can be configured to meet your organization's business needs.

There are many different topologies that can be configured; which range from a simple unidirectional topology to the more complex peer-to-peer. No matter the architecture, Oracle GoldenGate provides similarities between them, making administration easier.



For full information about processing methodology, supported topologies and functionality, and configuration requirements, see the Oracle GoldenGate documentation for your database.

Oracle GoldenGate Product Family

There are numerous products in the Oracle GoldenGate product family.

- **Oracle GoldenGate Veridata** : Oracle GoldenGate Veridata compares one set of data to another and identifies data that is out-of-sync, and allows you to repair any data that is found out-of-sync.
- **Oracle GoldenGate Plug-in for EMCC**: The Enterprise Manager Plug-in for Oracle GoldenGate extends the Oracle Enterprise Manager Cloud Control and

provides visual support for monitoring and managing Oracle GoldenGate processes.

- **Oracle GoldenGate Monitor:** Oracle GoldenGate Monitor is a real-time, Web-based monitoring console that delivers an at-a-glance, graphical view of all of the Oracle GoldenGate instances and their associated databases within your enterprise.
- **Oracle GoldenGate for Big Data:** Oracle GoldenGate for Big Data contains built-in support to write operation data from Oracle GoldenGate trail records into various Big Data targets (such as, HDFS, HBase, Kafka, Flume, JDBC, Cassandra, and MongoDB).
- **Oracle GoldenGate Application Adapters:** Oracle GoldenGate Application Adapters integrate with installations of the Oracle GoldenGate core product to bring in Java Message Service (JMS) information or to deliver information as JMS messages or files.
- **Oracle GoldenGate for HP NonStop (Guardian):** Oracle GoldenGate for HP NonStop enables you to manage business data at a transactional level by extracting and replicating selected data records and transactional changes across a variety of heterogeneous applications and platforms.
- **Oracle GoldenGate Studio:** Oracle GoldenGate Studio enables you to design and deploy high-volume, real-time replication by automatically handling table and column mappings, allowing drag and drop custom mappings, generating best practice configurations from templates, and contains context sensitive help.

2

Getting Started with Oracle GoldenGate

Oracle GoldenGate supports two architectures, the Classic Architecture and the Microservices Architecture (MA).

Oracle GoldenGate can be configured for the following purposes:

- A static extraction of data records from one database and the loading of those records to another database.
- Continuous extraction and replication of transactional Data Manipulation Language (DML) operations and data definition language (DDL) changes (for supported databases) to keep source and target data consistent.
- Data extraction from supported database sources and replication to Big Data and file targets using Oracle GoldenGate for Big Data.

Oracle GoldenGate Architectures Overview

The following table describes the two Oracle GoldenGate architectures and when you should use each of the architectures.

X	Classic Architecture	Microservices Architecture
What is it?	Oracle GoldenGate Classic Architecture is the original architecture for enterprise replication. This architecture provides the processes and files required to effectively transfer transactional data across a variety of topologies. These processes and files form the main components of the classic architecture and was the main product installation method until the Oracle GoldenGate 12c (12.3.0.1) release.	Oracle GoldenGate Microservices Architecture is a microservices architecture that enables REST services as part of the Oracle GoldenGate environment. The REST-enabled services provide API end-points that can be leveraged for remote configuration, administration, and monitoring through web-based consoles, an enhanced command line interface, PL/SQL and scripting languages.

X	Classic Architecture	Microservices Architecture
When should I use it?	<p>Oracle GoldenGate can be installed and configured to use the Oracle GoldenGate Classic Architecture only if MA release is not available for that platform, mentioned in the following scenarios:</p> <ul style="list-style-type: none"> • A static extraction of data records from one database and the loading of those records to another database. • Continuous extraction and replication of transactional Data Manipulation Language (DML) operations and Data Definition Language (DDL) changes (for supported databases) to keep source and target data consistent. • Extraction from a database and replication to a file outside the database. • Capture from heterogeneous database sources. 	<p>Oracle GoldenGate can be installed and configured to use the Oracle GoldenGate Microservices Architecture for the following purposes:</p> <ul style="list-style-type: none"> • Large scale and cloud deployments with fully-secure HTTPS interfaces and Secure WebSockets for streaming data. • Simpler management of multiple implementations of Oracle GoldenGate environments and control user access for the different aspects of Oracle GoldenGate setup and monitoring. • Support system managed database sharding to deliver fine-grained, multi-master replication where all shards are writable, and each shard can be partially replicated to other shards within a shardgroup. • Support the following features: <ul style="list-style-type: none"> – Thin and browser-based clients – Network security – User Authorization – Distributed deployments – Remote administration – Performance monitoring and orchestration – Coordination with other systems and services in an Oracle Database environment. – Custom embedding of Oracle GoldenGate into applications or to use secure, remote HTML5 applications.

X	Classic Architecture	Microservices Architecture
Which databases are supported?	Classic Architecture supports all supported databases as per the certification matrix .	MA only supports the Oracle database for an end-to-end MA-only topology. However, it is possible for a source Oracle GoldenGate Classic associated with heterogeneous databases to replicate to a target Oracle GoldenGate MA with Oracle, or a source Oracle GoldenGate MA with Oracle to replicate to a target Oracle GoldenGate legacy with heterogeneous databases.

Topics:

- [Oracle GoldenGate Supported Processing Methods and Databases](#)
- [Components of Oracle GoldenGate Microservices Architecture](#)
You can use Oracle GoldenGate Microservices Architecture to configure and manage your data replication using an HTML user interface.
- [Components of Oracle GoldenGate Classic Architecture](#)
You can use the Oracle GoldenGate Classic Architecture to configure and manage your data replications from the command line.
- [Common Data Replication Processes](#)
There are a number of data replication processes that are common to both Oracle GoldenGate architectures.

Oracle GoldenGate Supported Processing Methods and Databases

Oracle GoldenGate enables the exchange and manipulation of data at the transaction level among multiple, heterogeneous platforms across the enterprise. It moves committed transactions with transaction integrity and minimal overhead on your existing infrastructure. Its modular architecture gives you the flexibility to extract and replicate selected data records, transactional changes, and changes to DDL (data definition language) across a variety of topologies.

 **Note:**

Support for DDL, certain topologies, and capture or delivery configurations varies by the database type. See *Using Oracle GoldenGate for Oracle Database* and *Using Oracle GoldenGate for Heterogeneous Databases* for detailed information about supported features and configurations.

Here is a list of the supported processing methods.

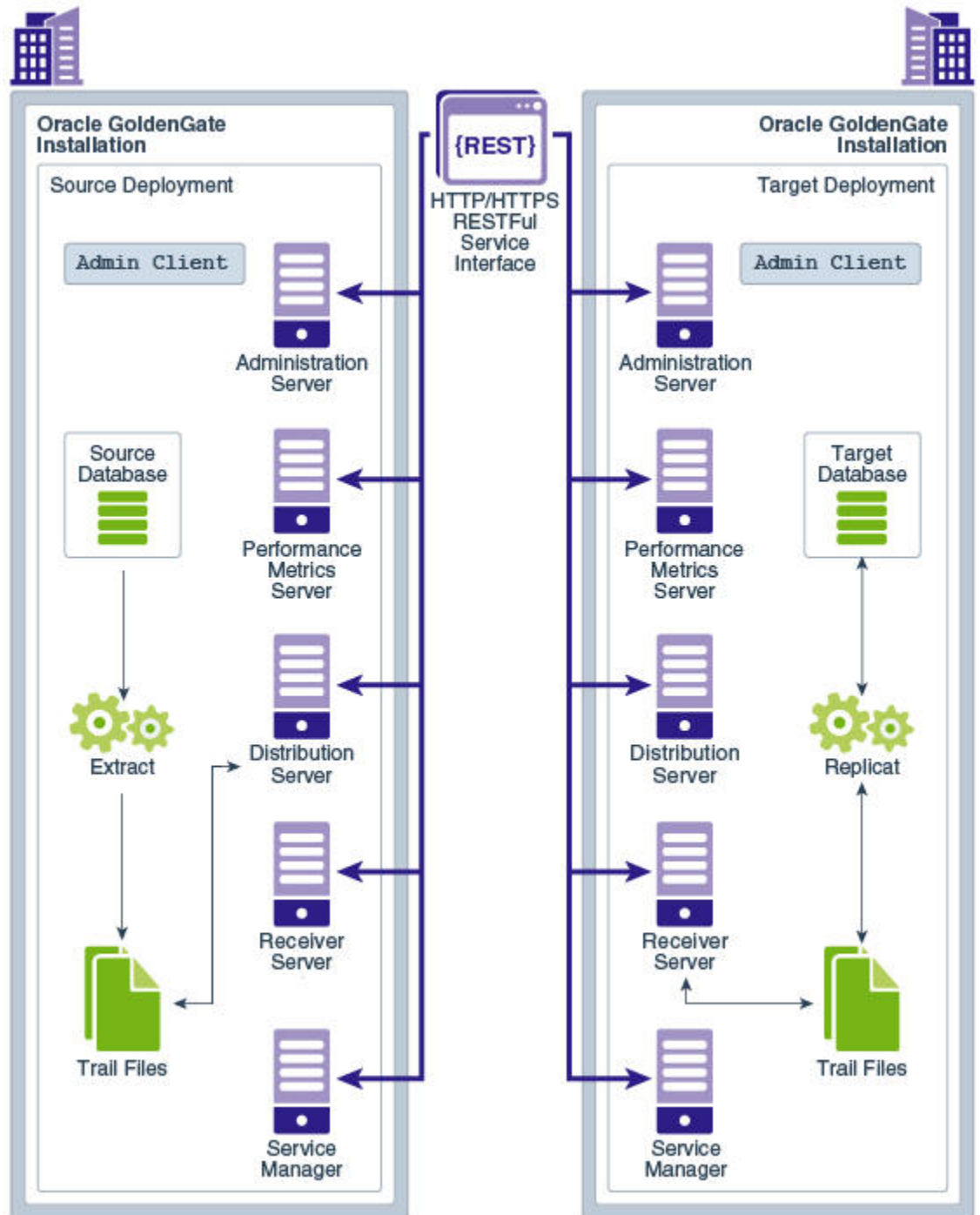
Database	Log-Based Extraction (capture)	Non-Log-Based Extraction ¹ (capture)	Replication (delivery)
DB2 for i	N/A	N/A	X
DB2 LUW	X	N/A	X
DB2 z/OS	X	N/A	X
Oracle Database	X	N/A	X
MySQL	X	N/A	X
SQL Server	N/A	X	X
Terradata	N/A	N/A	X

¹ Non-Log-Based Extraction uses a capture module that communicates with the Oracle GoldenGate API to send change data to Oracle GoldenGate.

Components of Oracle GoldenGate Microservices Architecture

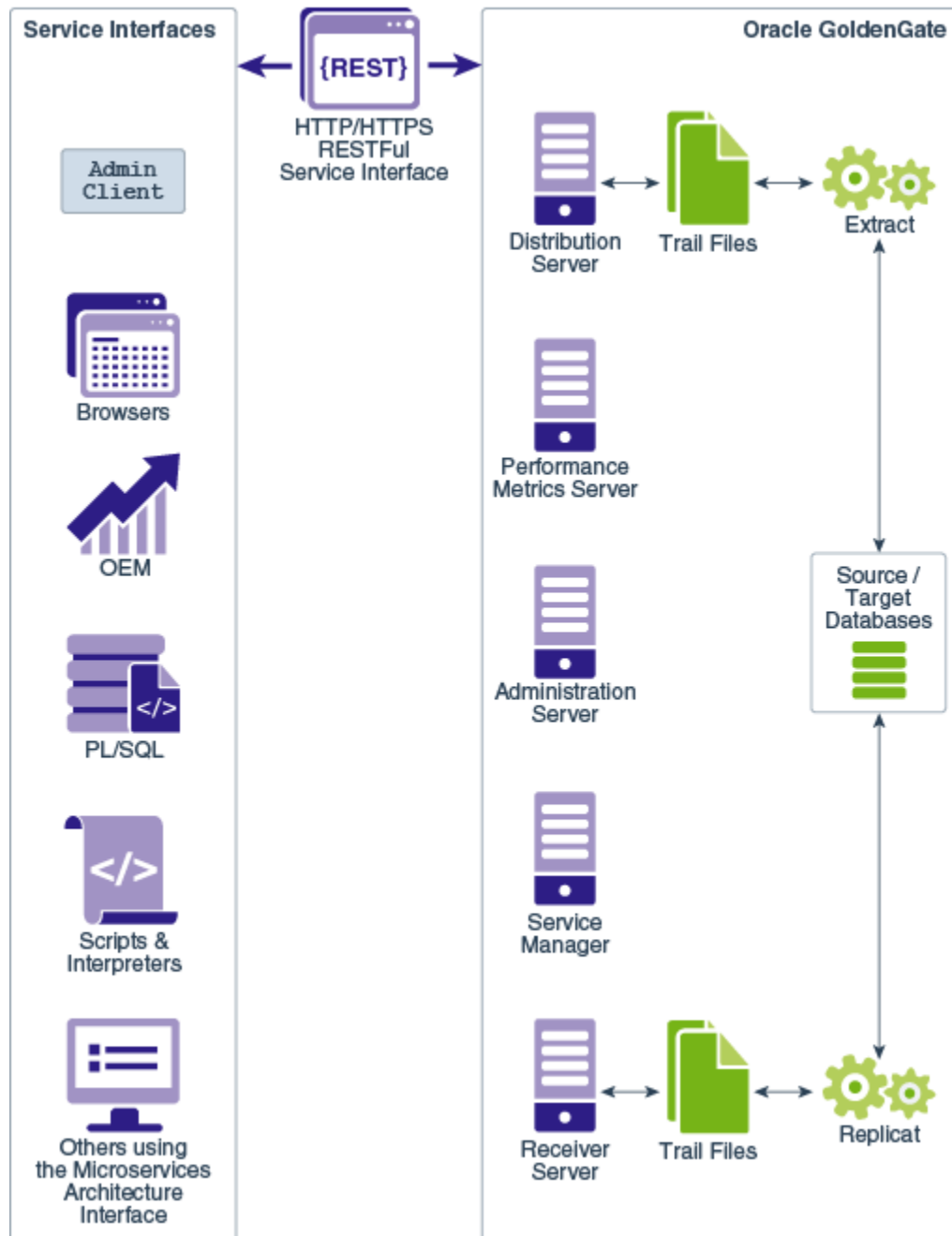
You can use Oracle GoldenGate Microservices Architecture to configure and manage your data replication using an HTML user interface.

There are five main components of the Oracle GoldenGate MA. The following diagram illustrates how replication processes operate within a secure REST API environment.



The Oracle GoldenGate MA provides all the tools you need to configure, monitor, and administer deployments and security. It is designed with the industry-standard HTTPS communication protocol and the JavaScript Object Notation (JSON) data interchange format. In addition, the architecture provides you with the ability to verify the identity of clients with basic authentication or Secure Sockets Layer client certificates.

The following diagram shows a variety of clients (Oracle products, command line, browsers, and programmatic REST API interfaces) that you can use to administer your deployments using the service interfaces.



Topics:

- [What is a Service Manager?](#)
A Service Manager acts as a watchdog for other services available with Microservices Architecture.
- [What is an Administration Server?](#)
The Administration Server supervises, administers, manages, and monitors processes within an Oracle GoldenGate deployment.
- [What is a Receiver Server?](#)
A Receiver Server is the central control service that handles all incoming trail files. It interoperates with the Distribution Server and provides compatibility with the classic architecture pump for remote classic deployments.

- [What is a Distribution Server?](#)
A Distribution Server is a service that functions as a networked data distribution agent in support of conveying and processing data and commands in a distributed deployment. It is a high performance application that is able to handle multiple commands and data streams from multiple source trail files, concurrently.
- [What is a Performance Metrics Server?](#)
To access the Performance Management Server APIs, you need the Oracle GoldenGate Management Pack Plug-in.
- [What is the Admin Client?](#)
The Admin Client is a command line utility (similar to the classic GGSCI utility). You can use it to issue the complete range of commands that configure, control, and monitor Oracle GoldenGate.
- [Roadmap for Implementing the Microservices Architecture](#)
Microservices Architecture is based on the REST API. After installing the Microservices Architecture, it is accessible through an HTML5 interface, REST APIs, and the command line.

What is a Service Manager?

A Service Manager acts as a watchdog for other services available with Microservices Architecture.

A Service Manager allows you to manage one or multiple Oracle GoldenGate deployments on a local host.

Optionally, Service Manager may run as a system service and maintains inventory and configuration information about your deployments and allows you to maintain multiple local deployments. Using the Service Manager, you can start and stop instances, and query deployments and the other services.

What is an Administration Server?

The Administration Server supervises, administers, manages, and monitors processes within an Oracle GoldenGate deployment.

The Administration Server operates as the central control entity for managing the replication components in your Oracle GoldenGate deployments. You use it to create and manage your local Extract and Replicat processes without having to have access to the server where Oracle GoldenGate is installed. The key feature of the Administration Server is the REST API service Interface that can be accessed from any HTTP or HTTPS client, such as the Microservices Architecture service interfaces or other clients like Perl and Python.

In addition, the Admin Client can be used to make REST API calls to communicate directly with the Administration Server, see [What is the Admin Client?](#)

The Administration Server is responsible for coordinating and orchestrating Extracts, Replicats, and paths to support greater automation and operational managements. Its operation and behavior is controlled through published query and service interfaces. These interfaces allow clients to issue commands and control instructions to the Administration Server using REST JSON-RPC invocations that support REST API interfaces.

The Administration Server includes an embedded web application that you can use directly with any web browser and does not require any client software installation.

Use the Administration Server to create and manage:

- Extract and Replicat processes
 - Add, alter, and delete
 - Register and unregister
 - Start and stop
 - Review process information, statistics, reports, and status including LAG and checkpoints
 - Retrieve the report and discard files
- Configuration (parameter) files
- Checkpoint, trace, and heartbeat tables
- Supplemental logging for procedural replication, schema, and tables
- Tasks both custom and standard, such as auto-restart and purge trails
- Credential stores
- Encryption keys (`MASTERKEY`)
- Add users and assign their roles

What is a Receiver Server?

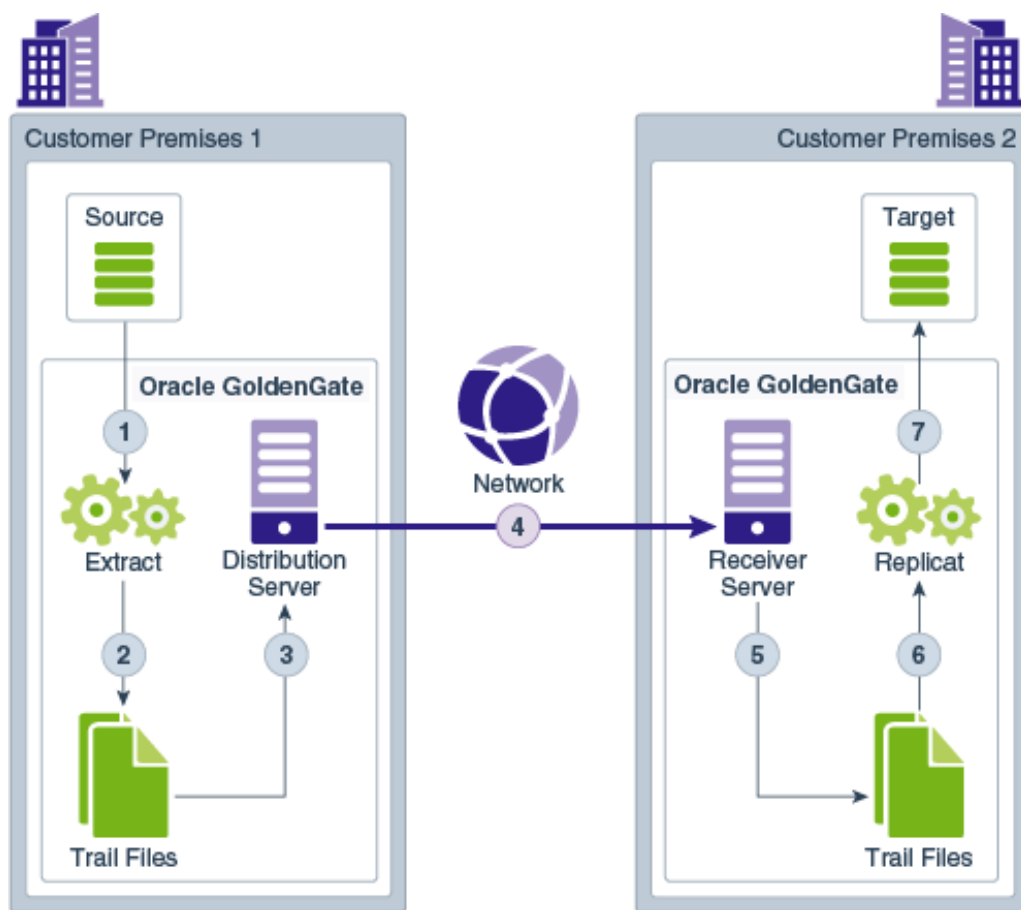
A Receiver Server is the central control service that handles all incoming trail files. It interoperates with the Distribution Server and provides compatibility with the classic architecture pump for remote classic deployments.

A Receiver Server replaces multiple discrete target-side Collectors with a single instance service.

Use Receiver Server to:

- Monitor path events
- Query the status of incoming paths
- View the statistics of incoming paths
- Diagnose path issues

WebSockets is the default HTTPS initiated full-duplex streaming protocol used by the Receiver Server. It enables you to fully secure your data using SSL security. The Receiver Server seamlessly traverses through HTTP forward and reverse proxy servers as illustrated in [#unique_26/unique_26_Connect_42_FIG-125-6797A798](#).



Additionally, the Receiver Server supports the following protocols:

- UDT—UDP-based protocol for wide area networks. For more information, see <http://udt.sourceforge.net/>.
- Classic Oracle GoldenGate protocol—For classic deployments so that the Distribution Server communicates with the Collector and the Data Pump communicates with the Receiver Server.

 **Note:**

TCP encryption does not work in a mixed environment of Classic and Microservices architecture. The Distribution Server in Microservices Architecture cannot be configured to use the TCP encryption to communicate with the Server Collector in Classic Architecture running in a deployment. Also, the Receiver Server in Microservices Architecture cannot accept a connection request from a data pump in Classic Architecture configured with `RMTHOST ... ENCRYPT` parameter running in a deployment.

What is a Distribution Server?

A Distribution Server is a service that functions as a networked data distribution agent in support of conveying and processing data and commands in a distributed deployment. It is a high performance application that is able to handle multiple commands and data streams from multiple source trail files, concurrently.

Distribution Server replaces the classic multiple source-side data pumps with a single instance service. This server distributes one or more trails to one or more destinations and provides lightweight filtering only (no transformations).

Multiple communication protocols can be used, which provide you the ability to tune network parameters on a per path basis. These protocols include:

- Oracle GoldenGate protocol for communication between the Distribution Server and the Collector in a non services-based (classic) target. It is used for interoperability.

Note:

TCP encryption does not work in a mixed environment of Classic and Microservices architecture. The Distribution Server in Microservices Architecture cannot be configured to use the TCP encryption to communicate with the Server Collector in Classic Architecture running in a deployment. Also, the Receiver Server in Microservices Architecture cannot accept a connection request from a data pump in Classic Architecture configured with `RMTHOST ... ENCRYPT` parameter running in a deployment.

- WebSockets for HTTPS-based streaming, which relies on SSL security.
- UDT for wide area networks.
- Proxy support for cloud environments:
 - SOCKS5 for any network protocol.
 - HTTP for HTTP-type protocols only, including WebSocket.
- Passive Distribution Server to initiate path creation from a remote site. **Paths** are source-to-destination replication configurations though are not included in this release.

Note:

There is no content transformation by this service.

What is a Performance Metrics Server?

To access the Performance Management Server APIs, you need the Oracle GoldenGate Management Pack Plug-in.

The Performance Metrics Server uses the metrics service to collect and store instance deployment performance results. This metrics collection and repository is separate from the administration layer information collection. You can monitor performance metrics using other embedded web applications and use the data to tune your deployments for maximum performance. All Oracle GoldenGate processes send metrics to the Performance Metrics Server. You can use the Performance Metrics Server in both Microservices Architecture and Classic Architecture.

Use the Performance Metrics Server to:

- Query for various metrics and receive responses in the services JSON format or the classic XML format
- Integrate third party metrics tools
- View error logs
- View active process status
- Monitor system resource utilization

What is the Admin Client?

The Admin Client is a command line utility (similar to the classic GGSCI utility). You can use it to issue the complete range of commands that configure, control, and monitor Oracle GoldenGate.

Admin Client is used to create, modify, and remove processes, rather than using MA. It's not used by MA services such as the Administration, Distribution and other servers. For example, you can either use Admin Client to execute all the commands necessary to create an Extract or customize a new Extract application, or use the Administration Server available with MA to configure an Extract.

 **Note:**

Ensure that the `OGG_HOME`, `OGG_VAR_HOME`, and `OGG_ETC_HOME` are set up correctly in the environment.

For more information on environment variables, see [Setting Environment Variables](#).

The way that you use the Admin Client while similar is different in some ways in support of the MA design:

GGSCI	Admin Client
Connects to local processes	Connects to any MA deployment
Requires local machine access, typically SSH	Requires HTTP or HTTPS access
Application logic executed locally	Application logic executed remotely
Requires connection to DBMS	No connection to DBMS required
Uses operating system security	Uses MA security
Authenticated and authorized once	Authenticated and authorized for each operation
No special connect semantics	Requires a <code>CONNECT</code> command

GGSCI	Admin Client
Supports USERID, PASSWORD, and USERIDALIAS	Supports USERIDALIAS only
REGISTER EXTRACT before ADD EXTRACT	REGISTER EXTRACT after ADD EXTRACT
Non-secure communications	Encrypted communications using SSL
Uses pump processes	Use Distribution Server

The Admin Client was designed with GGSCI as the basis. The following table describes the new, deleted, and deprecated commands in the Admin Client:

Table 2-1 Admin Client Commands

New Commands	Deleted Commands and Processes:	Deprecated Commands
CONNECT DISCONNECT [START STATUS STOP] SERVICE [ADD ALTER DELETE INFO [KILL START STATS STOP] [EDIT VIEW] GLOBALS CD	* MGR * JAGENT * CREATE DATASTORE SUBDIRS FC DUMPDDL INFO MARKER	ADD CREDENTIALSTORE [CREATE OPEN] WALLET

Roadmap for Implementing the Microservices Architecture

Microservices Architecture is based on the REST API. After installing the Microservices Architecture, it is accessible through an HTML5 interface, REST APIs, and the command line.

To start using the Microservices Architecture, the following must be accessible:

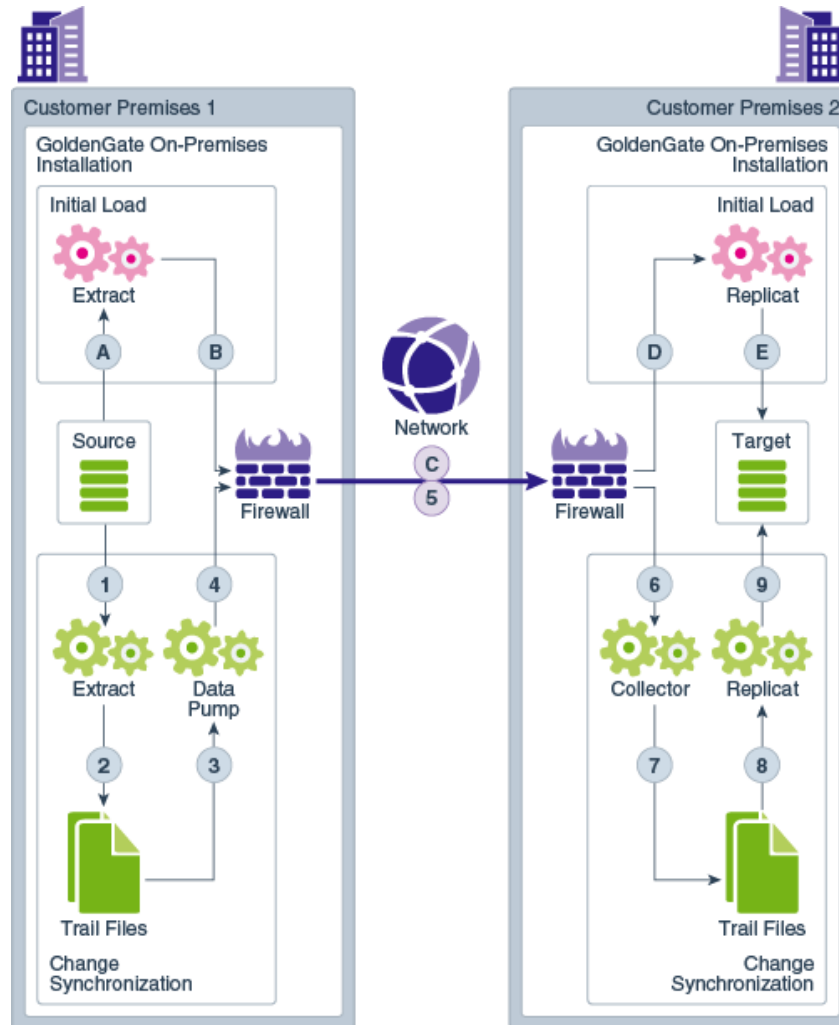
- The Database to which Oracle GoldenGate Microservices Architecture connects to.
- Oracle GoldenGate users must be configured.

This topic describes the roadmap for implementing Microservices Architecture components and clients.

Task	More Information
Installing MA	Installing the Microservices Architecture for Oracle Database
Starting the Service Manager	How to Start and Stop the Service Manager
Starting the Microservices	Quick Tour of the Service Manager Home Page
Starting the Processes	Quick Tour of the Administration Server Home Page
(Optional) Starting the Admin Client	How to Use the Admin Client

Components of Oracle GoldenGate Classic Architecture

You can use the Oracle GoldenGate Classic Architecture to configure and manage your data replications from the command line.



 **Note:**

This is the basic configuration. Depending on your business needs and use case, you can configure different variations of this model.

Topics:

- [What is a Manager?](#)
Manager is the control process of Oracle GoldenGate. Manager must be running on each system in the Oracle GoldenGate configuration before the Extract or Replicat processes can be started.

- [What is a Data Pump?](#)
Data pump is a secondary Extract group within the source Oracle GoldenGate configuration.
- [What is a Collector?](#)
The Collector is started by the manager process and is a process that runs in the background on the target system. It reassembles the transactional data into a target trail.
- [What is GGSCI?](#)
You can use the Oracle GoldenGate Software Command Interface (GGSCI) commands to create data replications. This is the command interface between you and Oracle GoldenGate functional components.

What is a Manager?

Manager is the control process of Oracle GoldenGate. Manager must be running on each system in the Oracle GoldenGate configuration before the Extract or Replicat processes can be started.

Manager must also remain running while the Extract and Replicat processes are running so that resource management functions are performed. One Manager process can control many Extract or Replicat processes.

Manager performs the following functions:

- Starts Oracle GoldenGate processes
- Starts dynamic processes
- Maintains port numbers for processes
- Purges Trail files based on retention rules
- Creates event, error, and threshold reports

One Manager process can control many Extract or Replicat processes. On Windows systems, Manager can run as a service. See olink:GWUAD-GUID-5005AF6D-76D2-4C72-80E2-AD33C24F0C26 for more information about the Manager process and configuring TCP/IP connections.

What is a Data Pump?

Data pump is a secondary Extract group within the source Oracle GoldenGate configuration.

If you configure a data pump, the Extract process writes all the captured operations to a trail file on the source database. The data pump reads the trail file on the source database and sends the data operations over the network to the remote trail file on the target database. Configuring a data pump is highly recommended for most configurations. If a data pump is not used, the Extract streams all the captured operations to a trail file on the remote target database. In a typical configuration with a data pump, however, the primary Extract group writes to a trail on the source system. The data pump reads this trail and sends the data operations over the network to a remote trail on the target. The data pump adds storage flexibility and also serves to isolate the primary Extract process from TCP/IP activity.

In general, a data pump can perform data filtering, mapping, and conversion

The data pump can be configured in two ways:

- Perform data manipulation: Data Pump can be configured to perform data filtering, mapping, and conversion.
- Perform no data manipulation: Data Pump can be configured in pass-through mode, where data is passively transferred as-is, without manipulation. Pass-through mode increases the throughput of the Data Pump, because all of the functionality that looks up object definitions is bypassed.

Though configuring a data pump is optional, Oracle recommends it for most configurations. Some reasons for using a data pump include the following:

- **Protection against network and target failures:** In a basic Oracle GoldenGate configuration, with only a trail on the target system, there is nowhere on the source system to store the data operations that Extract continuously extracts into memory. If the network or the target system becomes unavailable, Extract could run out of memory and abend. However, with a trail and data pump on the source system, captured data can be moved to disk, preventing the abend of the primary Extract. When connectivity is restored, the data pump captures the data from the source trail and sends it to the target system(s).
- **You are implementing several phases of data filtering or transformation.** When using complex filtering or data transformation configurations, you can configure a data pump to perform the first transformation either on the source system or on the target system, or even on an intermediary system, and then use another data pump or the Replicat group to perform the second transformation.
- **Consolidating data from many sources to a central target.** When synchronizing multiple source databases with a central target database, you can store extracted data operations on each source system and use data pumps on each of those systems to send the data to a trail on the target system. Dividing the storage load between the source and target systems reduces the need for massive amounts of space on the target system to accommodate data arriving from multiple sources.
- **Synchronizing one source with multiple targets.** When sending data to multiple target systems, you can configure data pumps on the source system for each target. If network connectivity to any of the targets fails, data can still be sent to the other targets.

What is a Collector?

The Collector is started by the manager process and is a process that runs in the background on the target system. It reassembles the transactional data into a target trail.

When the Manager receives a connection request from an Extract process, the Collector scans and binds to an available port and sends the port number to the Manager for assignment to the requesting Extract process. The Collector also receives the captured data that is sent by the Extract process and writes them to the remote trail file.

Collector is started automatically by the Manager when a network connection is required, so Oracle GoldenGate users do not interact with it. Collector can receive information from only one Extract process, so there is one Collector for each Extract that you use. Collector terminates when the associated Extract process terminates.

 **Note:**

Collector can be run manually, if needed. This is known as a static Collector (as opposed to the regular, dynamic Collector). Several Extract processes can share one static Collector; however, a one-to-one ratio is optimal. A static Collector can be used to ensure that the process runs on a specific port.

By default, Extract initiates TCP/IP connections from the source system to Collector on the target, but Oracle GoldenGate can be configured so that Collector initiates connections from the target. Initiating connections from the target might be required if, for example, the target is in a trusted network zone, but the source is in a less trusted zone.

What is GGSCI?

You can use the Oracle GoldenGate Software Command Interface (GGSCI) commands to create data replications. This is the command interface between you and Oracle GoldenGate functional components.

To start GGSCI, change directories to the Oracle GoldenGate installation directory, and then run the `ggsci` executable file.

 **Note:**

The environment variable `OGG_HOME` must be set before GGSCI can be started.

Common Data Replication Processes

There are a number of data replication processes that are common to both Oracle GoldenGate architectures.

Topics:

- [What is an Extract?](#)
Extract is a process that is configured to run against the source database or configured to run on a downstream mining database (Oracle only) with capturing data generated in the true source database located somewhere else. This process is the extraction or the data capture mechanism of Oracle GoldenGate.
- [What is a Trail?](#)
A trail is a series of files on disk where Oracle GoldenGate stores the captured changes to support the continuous extraction and replication of database changes.
- [What is a Replicat?](#)
Replicat is a process that delivers data to a target database. It reads the trail file on the target database, reconstructs the DML or DDL operations, and applies them to the target database.

What is an Extract?

Extract is a process that is configured to run against the source database or configured to run on a downstream mining database (Oracle only) with capturing data generated in the true source database located somewhere else. This process is the extraction or the data capture mechanism of Oracle GoldenGate.

You can configure an Extract for the following use cases:

- **Initial Loads:** When you set up Oracle GoldenGate for initial loads, the Extract process captures the current, static set of data directly from the source objects.
- **Change Synchronization:** When you set up Oracle GoldenGate to keep the source data synchronized with another set of data, the Extract process captures the DML and DDL operations performed on the configured objects after the initial synchronization has taken place. Extracts can run locally on the same server as the database or on another server using the downstream Integrated Extract for reduced overhead. It stores these operations until it receives commit records or rollbacks for the transactions that contain them. If it receives a rollback, it discards the operations for that transaction. If it receives a commit, it persists the transaction to disk in a series of files called a trail, where it is queued for propagation to the target system. All the operations in each transaction are written to the trail as a sequentially organized transaction unit and are in the order in which they were committed to the database (commit sequence order). This design ensures both speed and data integrity.

 **Note:**

Extract ignores operations on objects that are not in the Extract configuration, even though a transaction may also include operations on objects that are in the Extract configuration.

The Extract process can be configured to extract data from three types of data sources:

- **Source tables:** This source type is used for initial loads.
- **Database recovery logs or transaction logs:** While capturing from the logs, the actual method varies depending on the database type. An example of this source type is the Oracle Database redo logs.
- **Third-party capture modules:** This method provides a communication layer that passes data and metadata from an external API to the Extract API. The database vendor or a third-party vendor provides the components that extract the data operations and pass them to Extract.

What is a Trail?

A trail is a series of files on disk where Oracle GoldenGate stores the captured changes to support the continuous extraction and replication of database changes.

A trail can exist on the source system, an intermediary system, the target system, or any combination of those systems, depending on how you configure Oracle GoldenGate. On the local system, it is known as an Extract trail (or local trail). On a remote system, it is known as a remote trail. By using a trail for storage, Oracle

GoldenGate supports data accuracy and fault tolerance. The use of a trail also allows extraction and replication activities to occur independently of each other. With these processes separated, you have more choices for how data is processed and delivered. For example, instead of extracting and replicating changes continuously, you could extract changes continuously and store them in the trail for replication to the target later, whenever the target application needs them.

In addition, trails allow Oracle Database to operate in heterogeneous environment. The data is stored in a trail file in a consistent format, so it can be read by Replicat process for all supported databases. For more information, see About the Oracle GoldenGate Trail.

Processes that Write to the Trail File:

In Oracle GoldenGate Classic, the Extract and the data pump processes write to the trail. Only one Extract process can write to a given local trail. All local trails must have different full-path names though you can use the same trail names in different paths.

Multiple data pump processes can each write to a trail of the same name, but the physical trails themselves must reside on different remote systems, such as in a data-distribution topology. For example, a data pump named `pumpm` and a data pump named `pumpn` can both reside on `sys01` and write to a remote trail named `aa`. `Pumpm` can write to trail `aa` on `sys02`, while `pumpn` can write to trail `aa` on `sys03`.

In Oracle GoldenGate MA, Distribution Server and distribution paths are used to write the remote trail.

Processes that Read from the Trail File:

The data pump and Replicat processes read from the trail files. The data pump extracts DML and DDL operations from a local trail that is linked to an Extract process, performs further processing if needed, and transfers the data to a trail that is read by the next Oracle GoldenGate process downstream (typically Replicat, but could be another data pump if required).

The Replicat process reads the trail and applies the replicated DML and DDL operations to the target database.

Trail File Creation and Maintenance:

The trail files are created as needed during processing. You specify a two-character name for the trail when you add it to the Oracle GoldenGate configuration with the `ADD RMTTRAIL` or `ADD EXTTRAIL` command. By default, trails are stored in the `dirdat` sub-directory of the Oracle GoldenGate directory. You can specify a six or nine digit sequence number using the `TRAIL_SEQLEN_9D` | `TRAIL_SEQLEN_6D` GLOBALS parameter; `TRAIL_SEQLEN_9D` is set by default.

As each new file is created, it inherits the two-character trail name appended with a unique nine digit sequence number from 000000000 through 999999999 (for example `c:\ggs\dirdat\tr000000001`). When the sequence number reaches 999999999, the numbering starts over at 000000000, and previous trail files are overwritten. Trail files can be purged on a routine basis by using the Manager parameter `PURGEOLDEXTRACTS`.

You can create more than one trail to separate the data from different objects or applications. You link the objects that are specified in a `TABLE` or `SEQUENCE` parameter to a trail that is specified with an `EXTTRAIL` or `RMTTRAIL` parameter in the Extract parameter file. To maximize throughput, and to minimize I/O load on the system,

extracted data is sent into and out of a trail in large blocks. Transactional order is preserved.

Converting Existing Trails to 9 Digit Sequence Numbers

You can convert trail files from 6-digit to 9-digit checkpoint record for the named extract groups. Use `convchk` native command to convert to 9-digit trail by stopping your Extract gracefully then using `convchk` to upgrade as follows:

```
convchk extract trail seqlen_9d
```

Start your Extract

You can downgrade from a 9 to 6 digit trail with the same process using this `convchk` command:

```
convchk extract trail seqlen_6d
```

Note:

Extract Files: You can configure Oracle GoldenGate to store extracted data in an extract file instead of a trail. The extract file can be a single file, or it can be configured to roll over into multiple files in anticipation of limitations on file size that are imposed by the operating system. It is similar to a trail, except that checkpoints are not recorded. The file or files are created automatically during the run. The same versioning features that apply to trails also apply to extract files.

What is a Replicat?

Replicat is a process that delivers data to a target database. It reads the trail file on the target database, reconstructs the DML or DDL operations, and applies them to the target database.

The Replicat process uses dynamic SQL to compile a SQL statement once and then executes it many times with different bind variables. You can configure the Replicat process so that it waits a specific amount of time before applying the replicated operations to the target database. For example, a delay may be desirable to prevent the propagation of errant SQL, to control data arrival across different time zones, or to allow time for other planned events to occur.

For the two common uses cases of Oracle GoldenGate, the function of the Replicat process is as follows:

- **Initial Loads:** When you set up Oracle GoldenGate for initial loads, the Replicat process applies a static data copy to target objects or routes the data to a high-speed bulk-load utility.
- **Change Synchronization:** When you set up Oracle GoldenGate to keep the target database synchronized with the source database, the Replicat process applies the source operations to the target objects using a native database interface or ODBC, depending on the database type.

You can configure multiple Replicat processes with one or more Extract processes and Data Pumps in parallel to increase throughput. To preserve data integrity, each set of processes handles a different set of objects. To differentiate among Replicat processes, you assign each one a group name

If you don't want to use multiple Replicat processes, you can configure a single Replicat process in parallel, coordinated, integrated mode.

- Parallel mode Parallel Replicat supports all databases using the non-integrated option. Parallel Replicat only supports replicating data from trails with full metadata, which requires the classic trail format. It takes into account dependencies between transactions, similar to Integrated Replicat. The dependency computation, parallelism of the mapping and apply is performed outside the database so can be off-loaded to another server. The transaction integrity is maintained in this process. In addition, parallel replicat supports the parallel apply of large transactions by splitting a large transaction into chunks and applying them in parallel. See About Parallel Replicat.
- Coordinated mode is supported on all databases that Oracle GoldenGate supports. In coordinated mode, the Replicat process is threaded. One coordinator thread spawns and coordinates one or more threads that execute replicated SQL operations in parallel. A coordinated Replicat process uses one parameter file and is monitored and managed as one unit. See About Coordinated Replicat Mode for more information.
- Integrated mode is supported for Oracle Database releases 11.2.0.4 or later. In integrated mode, the Replicat process leverages the apply processing functionality that is available within the Oracle Database. Within a single Replicat configuration, multiple inbound server child processes known as apply servers apply transactions in parallel while preserving the original transaction atomicity. See About Integrated Replicat for more information about integrated mode.

You can delay Replicat so that it waits a specific amount of time before applying the replicated operations to the target database. A delay may be desirable, for example, to prevent the propagation of errant SQL, to control data arrival across different time zones, or to allow time for other planned events to occur. The length of the delay is controlled by the `DEFERAPPLYINTERVAL` parameter.

3

Oracle GoldenGate Processes and Key Terms

Oracle GoldenGate has common data replication processes and architecture-specific processes as well.

Specific components of Classic Architecture and Microservices Architecture are discussed in [Components of Classic Architecture](#) and [Components of Microservices Architecture](#). However, there are various processes and key terms that are common to both the architectures of Oracle GoldenGate.

- [Overview of Target-Initiated Paths](#)

- [Oracle GoldenGate Key Terms and Concepts](#)

Apart from the two architectures and their components, there are some key terms that you should get familiar with.

Overview of Target-Initiated Paths

Target-initiated paths for microservices enable the Receiver Server to initiate a path to the Distribution Service on the target deployment and pull trail files. This feature allows the Receiver Server to create a target initiated path for environments such as Demilitarized Zone Paths (DMZ) or Cloud to on-premise, where the Distribution Server in the source Oracle GoldenGate deployment cannot open network connections in the target environment to the Receiver Server due to network security policies.

If the Distribution Server cannot initiate connections to the Receiver Server, but Receiver Server can initiate a connection to the machine running the Distribution Server, then the Receiver Server establishes a secure or non-secure target initiated path to the Distribution Server through a firewall or Demilitarized (DMZ) zone using Oracle GoldenGate and pull the requested trail files.

The Receiver Server end-points display that the retrieval of the trail files was initiated by the Receiver Server, see [Quick Tour of the Receiver Server Home Page](#).

You can enable this option from the Configuration Assistant wizard Security options, see [How to Create Deployments](#). For steps to create a target-initiated distribution path, see [How to Add a Target-Initiated Distribution Path in *Using the Oracle GoldenGate Microservices Architecture*](#).

Oracle GoldenGate Key Terms and Concepts

Apart from the two architectures and their components, there are some key terms that you should get familiar with.

Topics:

- [Overview of Process Types](#)
- [Overview of Groups](#)

- [Overview of the Commit Sequence Number \(CSN\)](#)

Overview of Process Types

Depending on the requirement, Oracle GoldenGate can be configured with the following processing types.

- An *online* Extract or Replicat process that runs until stopped by a user. Online processes maintain recovery checkpoints so that processing can resume after interruptions. You use online processes to continuously extract and replicate DML and DDL operations (where supported) to keep source and target objects synchronized. The `EXTRACT` and `REPLICAT` parameters apply to this process type.
- A *source-is-table* (also known as in *initial-load* Extract) Extract process extracts a current set of static data directly from the source objects in preparation for an initial load to another database. This process type does not use checkpoints. The `SOURCEISTABLE` parameter applies to this process type.
- A *special-run* Replicat process applies data within known begin and end points. You use a special-run Replicat for initial data loads, and it also can be used with an online Extract to apply data changes from the trail in batches, such as once a day rather than continuously. This process type does not maintain checkpoints because the run can be started over with the same begin and end points. The `SPECIALRUN` parameter applies to this process type.
- A *remote task* is a special type of initial-load process in which Extract communicates directly with Replicat over TCP/IP. Neither a Collector process nor temporary disk storage in a trail or file is used. The task is defined in the Extract parameter file with the `RMTTASK` parameter.

Overview of Groups

To differentiate among multiple Extract or Replicat processes on a system, you define processing *groups*. For example, to replicate different sets of data in parallel, you would create two Replicat groups.

A processing group consists of a process (either Extract or Replicat), its parameter file, its checkpoint file, and any other files associated with the process. For Replicat, a group may also include an associated checkpoint table. You define groups by using the `ADD EXTRACT` and `ADD REPLICAT` commands in the Oracle GoldenGate command interface, GGSCI.

All files and checkpoints relating to a group share the name that is assigned to the group itself. Any time that you issue a command to control or view processing, you supply a group name or multiple group names by means of a wildcard.

Overview of the Commit Sequence Number (CSN)

When working with Oracle GoldenGate, you might need to refer to a *Commit Sequence Number*, or CSN. A CSN is an identifier that Oracle GoldenGate constructs to identify a transaction for the purpose of maintaining transactional consistency and data integrity. It uniquely identifies a point in time in which a transaction commits to the database.

The CSN can be required to position Extract in the transaction log, to reposition Replicat in the trail, or for other purposes. It is returned by some conversion functions and is included in reports and certain GGSCI output.

A CSN is a monotonically increasing identifier generated by Oracle GoldenGate that uniquely identifies a point in time when a transaction commits to the database. Its purpose is to ensure transactional consistency and data integrity as transactions are replicated from source to target. Each kind of database management system generates some kind of unique serial number of its own at the completion of each transaction, which uniquely identifies the commit of that transaction. For example, the Oracle RDBMS generates a System Change Number, which is a monotonically increasing sequence number assigned to every event by Oracle RDBMS. The CSN captures this same identifying information and represents it internally as a series of bytes, but the CSN is processed in a platform-independent manner. A comparison of any two CSN numbers, each of which is bound to a transaction-commit record in the same log stream, reliably indicates the order in which the two transactions completed.

The CSN is cross-checked with the transaction ID (displayed as XID in Oracle GoldenGate informational output). The XID-CSN combination uniquely identifies a transaction even in cases where there are multiple transactions that commit at the same time, and thus have the same CSN. For example, this can happen in an Oracle RAC environment, where there is parallelism and high transaction concurrency.

The CSN value is stored as a token in any trail record that identifies the commit of a transaction. This value can be retrieved with the @GETENV column conversion function and viewed with the Logdump utility.

See About the Commit Sequence Number in *Administering Oracle GoldenGate* for more information about the CSN and a list of CSN values per database.